PR #29374 完整报告

sgl-project/sglang

[NPU] Fix mllama cross-attention crash in ascend extend SDPA

合并时间：2026-06-26 17:13

原文链接：http://prhub.com.cn/sgl-project/sglang/pull/29374

执行摘要

- 一句话：修复 NPU 上 mllama 交叉注意力崩溃问题
- 推荐动作：该 PR 值得关注，因为它展示了在支持不同注意力模式（自注意力 vs 交叉注意力）时，如何小心处理通用逻辑中的特殊分支。对于类似场景，建议增加单元测试以覆盖多种注意力组合，防止未来回归。

功能与动机

PR #26147 (Gemma4 SWA 支持）引入了一个回归问题，导致 mllama (Llama-3.2-11B-Vision-Instruct) 在 Ascend NPU 后端上交叉注意力崩溃，服务器无法启动。根本原因是 SWA 变更将冗余 query 张量大小从 seq_lens[seq_idx]（文本序列长度）改为 atten_end_kv - atten_start_kv (KV 窗口大小），这对 SWA 自注意力是正确的，但破坏了交叉注意力：在交叉注意力中，Q 来自文本 (16 tokens)，而 KV 来自编码器 / 图像 (6404 tokens)，冗余 Q 张量被设为编码器长度，但实际只有 16 个 query，导致形状不匹配。

实现拆解

在文件 python/sglang/srt/hardware_backend/npu/attention/ascend_torch_native_backend.py 的函数 run_sdpa_forward_extend 中，对冗余 query 张量的构建逻辑进行分场景处理：

1. 提取 is_swa_self_attn 标志位，代替原有的条件判断，明确区分 SWA 自注意力场景。
2. 对于 SWA 自注意力，维持原有行为：redundant_len = atten_end_kv - atten_start_kv，且 query_start_idx = max(prefill_seq_len_q - atten_start_kv, 0)，以保持 Q 与 K 长度一致用于 causal mask。
3. 对于交叉注意力或非 SWA 自注意力，还原为原始逻辑：redundant_len = int(seq_lens[seq_idx].item())，且 query_start_idx = prefill_seq_len_q，确保冗余 Q 张量大小与文本序列长度匹配，与 KV（编码器）长度解耦。
4. 这样既保证了 SWA gemma4 自注意力行为不变，又恢复了 mllama 交叉注意力的正确性。

关键文件：

- python/sglang/srt/hardware_backend/npu/attention/ascend_torch_native_backend.py（模块 NPU 注意力层；类别 source；类型 core-logic）：核心修复文件，修改了 run_sdpa_forward_extend 函数中的冗余 query 张量创建逻辑，分场景处理 SWA 自注意力和交叉注意力。

关键符号：run_sdpa_forward_extend

关键源码片段

python/sglang/srt/hardware_backend/npu/attention/ascend_torch_native_backend.py

核心修复文件，修改了 run_sdpa_forward_extend 函数中的冗余 query 张量创建逻辑，分场景处理 SWA 自注意力和交叉注意力。

```
# 根据注意力类型选择冗余 query 的尺寸
is_swa_self_attn = (
    sliding_window_size is not None
    and sliding_window_size > -1
    and encoder_lens is None
)
if is_swa_self_attn:
    # SWA 自注意力：需要与裁剪后的 KV 窗口匹配
    redundant_len = atten_end_kv - atten_start_kv
    query_start_idx = max(prefill_seq_len_q - atten_start_kv, 0)
else:
    # 交叉注意力 / 非 SWA 自注意力：使用原始文本序列长度，
    # 因为 Q（文本）和 KV（编码器）长度不同
    redundant_len = int(seq_lens[seq_idx].item())
    query_start_idx = prefill_seq_len_q

per_req_query_redundant = torch.zeros(
    (per_req_query.shape[0], redundant_len, per_req_query.shape[2]),
    dtype=per_req_query.dtype,
    device=per_req_query.device,
)
per_req_query_redundant[:, query_start_idx:, :] = per_req_query
```

评论区精华

此 PR 没有 review 评论讨论。审核人 sglang-npu-bot 直接批准了合并。

- 暂无高价值评论线程

风险与影响

- 风险：风险较低。变更仅涉及一个文件中单个函数内的分支逻辑，且已通过 MMMU 精度测试 (score=0.3301) 验证 mllama 行为。但缺少针对交叉注意力和 SWA 自注意力的单元测试覆盖，未来若修改相关逻辑可能再次引入回归。
- 影响：直接影响使用 Ascend NPU 后端的 mllama 模型（Llama-3.2-11B-Vision-Instruct）的交叉注意力功能，使其能够正常启动和推理。对 SWA gemma4 自注意力无影响。其他后端（如 CUDA）不受影响。影响范围小，但修复了关键的功能崩溃问题。
- 风险标记：缺少单元测试覆盖

关联脉络

- PR #26147 Gemma4 SWA support: 根因 PR，其修改导致了当前修复的回归问题。